

ltxfileinfo print version information for a LaTeX file

doc generated from the script with `gendoc`

bash script, version=2.06

Synopsis

```
ltxfileinfo [options] filename
```

Options

```
-H|--help      print full documentation via less and exit
-d|--date      print file's date only
-v|--version   print file's version only
-i|--info      print file's description text only
-l|--location  print file's full path only (same output as kpsewhich)
-f|--flat      output as 1 line with 4 tab-separated file, date,
               version and info fields
-D|--debug     for debugging, keep temporary files
```

Description

ltxfileinfo displays version information for LaTeX files. If no path information is given, the file is searched using `kpsewhich`. As an extra, for developers, the script will (use the `--star` or `--color` options) check the validity of the `\Provides...` statement in the files. The script uses code from Uwe Lück's `readprov.sty`.

Without an option, the output will be of the form:

```
$ ltxfileinfo ctable.sty
name: ctable.sty
date: 2012/08/23
vers: v1.24
info: Easy, key=value directed, option-rich, typesetting of floats
loca: /usr/local/texlive/2012/./texmf-local/tex/latex/ctable/ctable.sty
```

Missing information is represented by `--`:

```
$ ltxfileinfo yhmath.sty
name: yhmath.sty
date: --
vers: --
info: --
loca: /usr/local/texlive/2012/texmf-dist/tex/latex/yhmath/yhmath.sty
```

ltxfileinfo tries to detect (and report) mal-formatted dates and versions; use the `--star` option to mark such fields with a star, or use the `--color` option to color invalid fields red, if you work in a terminal with ANSI-coloring capabilities. These options are useful for developers who want to check the correctness of their `\Provides...` statements. For example:

```
$ ltxfileinfo -s arfonts.sty
file: ARfonts.sty* (case difference in \Provides statement)
date: 2006/01/01
vers: --*
info: Part of the Arabi package
loca: /usr/local/texlive/2012/texmf-dist/tex/latex/arabi/arfonts.sty
```

We see here, that the `\ProvidesPackage` statement has an incorrect first argument and has no version information.

Other output formats

With the `--date` option, only the file's date will be shown, unlabeled. The `--version`, `--location`, `--info` options are treated analogously.

```
$ ltxfileinfo -v chronology.sty
v1.0
```

The `--flat` option prints the fields (except `locat:`, the last field) on one line, unlabeled and tab-separated:

```
$ ltxfileinfo --star --flat chronology.sty
chronology.sty      2010/6/12*      v1.0      Horizontal timeline
```

The date is starred, because its format is not `yyyyImddd` (and the `--star` option is given.)

Bugs

On my system, I have a total of 8699 kpsewhich-detectable files that contain a `\Provides...` statement. I ran them all through `ltxfileinfo` and made the following summary of detected errors:

```
8695 Total \Provides... containing files tested

660 \Provides... could not be interpreted; reason:
160 Argument of \Provides... not equal to file's name
  8 Latex3 package (not handled yet)
480 Unidentified problem with \ProvidesPackage statement
 12 \Provides... used in file without a ... extension

8035 Files could be evaluated
482 had no date
 95 had a mal-formatted date
2666 had no version
357 had a mal-formatted version
765 had a \Provides... first argument different from the filename
```

The `.dtx` files have more problems than other files:

```
1345 .dtx files:

320 \Provides... could not be interpreted; reason:
 81 Argument of \Provides... not equal to...
  6 Latex3 package (not handled yet)
233 Unidentified problem with \ProvidesPackage statement

1025 Files could be evaluated
 49 had no date
 15 had a mal-formatted date
 99 had no version
 52 had a mal-formatted version
468 had a \Provides... first argument different from the filename
```

Author

Wybo Dekker

Copyright

Released under the [GNU General Public License](#)

Functions used:

excheck

```
synopsis:    excheck executable1 [executable2...]
description: check if all needed execs are there and getopt is GNU
```

handle_options

synopsis: handle_options "\$@"
description: handle the options.
globals used: Myname Version
globals set: args debug flat key mark1 mark2
returns: the number of remaining arguments